



PATENT ABSTRACTS OF JAPAN

(11) Publication number: 2000339152 A

(43) Date of publication of application: 08.12.00

(51) Int. Cl.

G06F 9/06

(21) Application number: 11149475

(22) Date of filing: 28.05.99

(71) Applicant: HITACHI DENSHI LTD

(72) Inventor: HIGURE KINICHI
KOBAYASHI TAKEHIKO

(54) METHOD FOR CHANGING APPLICATION PROGRAM

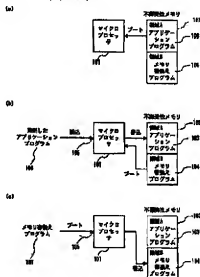
(57) Abstract:

PROBLEM TO BE SOLVED: To change a memory rewriting program while ROM is directly mounted as it is by writing the memory rewriting program into a non-volatile memory and booting the non-volatile memory.

SOLUTION: A non-volatile memory 102 is divided into two areas A and B, and the area A is allocated to an application program 103 and the area B to a memory rewriting program 104. For rewriting the application program 103, the memory rewriting program 104 written into the area B of the non-volatile memory 102 is booted and the application program 103 is rewritten. For rewriting the memory rewriting program 104, a memory rewriting program 107 is booted from the outer interface 105 of a micro processor 101 and the program is written into the area B of the non-volatile memory 102. Thus, the

memory rewriting program 104 is rewritten.

COPYRIGHT: (C)2000, JPO



(51) Int. Cl. ⁷	識別記号	F I	チーコード* (参考)
G 0 6 F 9/06	5 4 0	G 0 6 F 9/06	5 4 0 M 5 B 0 7 6
	4 1 0		4 1 0 S

審査請求 未請求 請求項の数2 O L (全 6 頁)

(21) 出願番号	特願平11-149475	(71) 出願人	000005429 日立電子株式会社 東京都千代田区神田和泉町1番地
(22) 出願日	平成11年5月28日 (1999.5.28)	(72) 発明者	日暮 敏一 東京都小平市御幸町32番地 日立電子株式会社小金井工場内
		(72) 発明者	小林 岳彦 東京都小平市御幸町32番地 日立電子株式会社小金井工場内
		Fターム (参考)	5B076 BA03 BA05 EB01

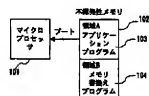
(54) 【発明の名称】 アプリケーションプログラム変更方法

(57) 【要約】

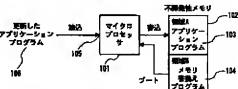
【課題】 マイクロプロセッサが読み実行するプログラムを、電気的に消去・書き換えが可能な不揮発性メモリから読み実行するブート方法においては、不揮発性メモリを書き換えるためのプログラムが必要になるが、本発明の課題はこのプログラムを、メモリを直接実装する場合でも、変更可能にすることにある。

【解決手段】 本発明では、不揮発性メモリを2つの領域に分割し、分割した片方の領域にメモリ書換えプログラムを格納し、このプログラムによりもう片方の領域へアプリケーションプログラムを、書き込む方法または、メモリ書換えプログラムをマイクロプロセッサの外部インターフェースから読み込む。

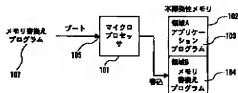
(a) 通常ブート



(b) アプリケーションプログラムの書き換え



(c) メモリ書換えプログラムの書き換え



【特許請求の範囲】

【請求項1】 実行するアプリケーションプログラムを、電気的に消去及び書換え可能な不揮発性メモリから読み込んで実行するマイクロプロセッサであって、前記マイクロプロセッサが、前記アプリケーションプログラムを更新する場合に、

該更新するアプリケーションプログラムを読み込み、前記不揮発性メモリに、前記アプリケーションプログラムを書換えまたは書込み処理を行う方法において、前記マイクロプロセッサに書込みまたは書換え処理を行わせるためのメモリ書換えプログラムを前記不揮発性メモリに記憶させることを特徴とするマイクロプロセッサのアプリケーションプログラム変更方法。

【請求項2】 実行するアプリケーションプログラムを、電気的に消去及び書換え可能な不揮発性メモリから読み込んで実行するマイクロプロセッサであって、前記マイクロプロセッサが、前記アプリケーションプログラムを更新する場合に、

該更新するアプリケーションプログラムを読み込み、前記不揮発性メモリに、前記アプリケーションプログラムを書換えまたは書込み処理を行う方法において、前記マイクロプロセッサに書込みまたは書換え処理を行わせるためのメモリ書換えプログラムを前記更新するアプリケーションプログラムと結合し、該結合したプログラムを読み込んで、前記不揮発性メモリに書換えまたは書込みを行うことを特徴とするマイクロプロセッサのアプリケーションプログラム変更方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、マイクロコンピュータやデジタルシグナルプロセッサなどを含む、マイクロプロセッサの実行プログラムを読み込む方法に関するものである。

【0002】

【従来の技術】マイクロプロセッサはプログラムを読み込み、読み込んだプログラムを実行する。その場合に、そのプログラムを電気的に消去または書換えが可能な不揮発性メモリ（以下、“プログラムを電気的に消去または書換えが可能な不揮発性メモリ”を“不揮発性メモリ”と称する）からブートし、また不揮発性メモリへのプログラムの書込みもマイクロプロセッサ自身が行う方法がある。その一例を以下に述べる。ここで、ブートとは、マイクロプロセッサがプログラムを読み込み、読み込んだプログラムを実行することをいう。

【0003】まず図3を用いて、マイクロプロセッサ自身が不揮発性メモリへの書込みを行うときに、必要な書込みの処理動作を行うプログラムをROM（Read Only Memory）からブートする方法について説明する。

【0004】図3は不揮発性メモリに書込まれたアプリケーションプログラムをブートする場合のデータの流れ

を示した図である。通常は、図3(a)に示すように、不揮発性メモリ102からアプリケーションプログラム103をブートする。このときメモリ書換えプログラム302が格納されているROM301とマイクロプロセッサ101とのバスは接続されていない。今、アプリケーションプログラム103を書換える場合、または実装した不揮発性メモリ102にアプリケーションプログラム103が書込まれていない場合の書込み方法を図3(b)によって説明する。図3(b)において、マイクロプロセッサ101はROM301からメモリ書換えプログラム302をブートする。このメモリ書換えプログラム302は更新したアプリケーションプログラム106をマイクロプロセッサ101の外部インターフェース105から読み込み、これを不揮発性メモリ102へ書込むプログラムであり、ブートすることによりアプリケーションプログラム103が書換えられる。

【0005】しかし、ROM301に書込まれているメモリ書換えプログラム302の内容を、不具合等の理由によって書換える場合、またはROM301にプログラムが書込まれていない場合に、プログラムを書換えたり、書込んだりすることはできない。

【0006】

【発明が解決しようとする課題】前述の従来技術には、アプリケーションプログラムの変更は容易だが、ROMを書込むメモリ書換えプログラムは、一度ROMを実装してしまうとメモリ書換えプログラムに不具合があっても変更することができない。

【0007】また、ROMを簡単に着脱できるように、ICソケットを取付け、ICソケットを介して実装すると、取外したROMに対してメモリ書換えプログラムを変更することが可能であるが、装置の小型化には適さない欠点があった。

【0008】本発明の目的は、上記のような欠点を除去し、ROMを直接実装したままで、メモリ書換えプログラムの変更を可能にすることにある。

【0009】

【課題を解決するための手段】上記目的を達成するために、本発明は、メモリ書換えプログラムを不揮発性メモリに書込み、その不揮発性メモリからブートすることにより実現した。

【0010】また本発明の他の方法として、メモリ書換えプログラムをマイクロプロセッサの外部インターフェースからブートすることにより実現した。

【0011】

【発明の実施の形態】まず、メモリ書換えプログラムを不揮発性メモリに書込み、その不揮発性メモリからブートする方法を図1を用いて説明する。図1は不揮発性メモリの領域の一部にメモリ書換えプログラムが格納されている場合のデータの流れを示した図である。図1は不揮発性メモリ102を2つの領域（領域Aと領域B）に分け、領域Aをアプリケーションプログラム103に、領域Bをメ

メモリ書換えプログラム104に割当てている。

【0012】通常は図1(a)に示すように、不揮発性メモリ102の領域Aからアプリケーションプログラム103をブートする。このとき、領域Bに格納されているメモリ書換えプログラム104にはアクセスされない。アプリケーションプログラム103を書換える場合は、図1(b)に示すように、不揮発性メモリ102の領域Bに書込まれているメモリ書換えプログラム104をブートし、アプリケーションプログラム103を書換える。このメモリ書換えプログラム104は更新したアプリケーションプログラム106をマイクロプロセッサ101の外部インターフェース105を介して読み込み、不揮発性メモリ102の領域Aに書込むことによって領域Aのアプリケーションプログラム103を更新する。メモリ書換えプログラム104を書換える場合、または実装後、不揮発性メモリ102の領域Bにメモリ書換えプログラムが書込まれていない場合には、図1(c)に示すように、メモリ書換えプログラム107をマイクロプロセッサ101の外部インターフェース105よりブートし、不揮発性メモリ102の領域Bに書込むことによって、メモリ書換えプログラム104を書換えるかまたは、片込む。このメモリ書換えプログラム107はメモリ書換えプログラム104を不揮発性メモリ102に書込むプログラムである。

【0013】次に、メモリ書換えプログラムをマイクロプロセッサの外部インターフェースからブートする方法を図2を用いて説明する。図2はメモリ書換えプログラムを外部外部からブートする場合のデータの流れを示した図である。図2は不揮発性メモリ102に、アプリケーションプログラム103だけを格納している。

【0014】通常の場合には、図2(a)に示すように、不揮発性メモリ102からアプリケーションプログラム103をブートする。不揮発性メモリ102の中のアプリケーションプログラム103を書換える場合は、メモリ書換えプログラム202（実行プログラム）にアプリケーションプログラム203（データテーブル）を結合したプログラム201をマイクロプロセッサ101が外部インターフェース105からブートする。このブートしたプログラム201は、データテーブルとして結合したアプリケーションプログラム203を不揮発性メモリ102に書込むものであり、これによって、不揮発性メモリ102が書換えられる。

【0015】以下、メモリ書換えプログラムを、不揮発性メモリに置く場合の一実施例を図4によって説明する。図4は、メモリ書換えプログラムを不揮発性メモリに置いた場合の一実施例を説明する図である。マイクロプロセッサとしてデジタルシグナルプロセッサ（DSP: Digital Signal Processor）401を用い、不揮発性メモリとしてフラッシュメモリ402を用いている。DSP401とフラッシュメモリ402はデータバス403とアドレスバス404で接続され、フラッシュメモリ402からのアドレスバス404のうち、上位ビットをレジスタ410に接続し、フラッシュメモリ402を2つの領域に分割して使用する。

【0016】レジスタ410は、DSP401のブート時に、フラッシュメモリ領域選択スイッチ406の値がサンプリングし、ブート後はDSP401から変更することができる。フラッシュメモリ402のアドレスバスのビット数をNとすると、フラッシュメモリ402のアドレス空間は0番地から 2^N-1 番地となる。図5(a)に示すように、フラッシュメモリ402を、0番地から $2^{N-1}-1$ 番地までの領域A501と 2^{N-1} 番地から 2^N-1 番地までの領域B502に2等分する。レジスタ410の内容が“0”の場合は、アドレスバス404の最上位ビットは“0”となり、図5(b)に示すように、DSP401からは領域A501が0番地から $2^{N-1}-1$ 番地と見て見られる。“1”の場合は、アドレスバス404の最上位ビットが“1”となり、図5(c)に示すように、領域B502が0番地から $2^{N-1}-1$ 番地と見て見られるため、領域を選択して使用することができる。

【0017】また、図4において、DSP401にブート選択スイッチ405を接続し、シリアルポート407からブートするか、またはフラッシュメモリ402からブートするかを選択する。図9はブート選択スイッチ405と領域選択スイッチ406の組合わせに対するDSP401のポート先を示したもので、ブート選択スイッチ405が“0”の場合、領域選択スイッチ406が“0”ならばフラッシュメモリ402の領域Aから、領域選択スイッチ406が“1”ならば領域Bからブートし、ブート選択スイッチ405が“1”の場合、領域選択スイッチ406の値に関わらずシリアルポートブートを行う。

【0018】フラッシュメモリ402の内容を書換える場合は、DSP401のシリアルポートインターフェース407をパーソナルコンピュータ408のシリアルポートインターフェース409に接続し、シリアルポートブートを行うプログラムおよびフラッシュメモリ402に書込むプログラムをパーソナルコンピュータ408から転送する。

【0019】図6は、図4の実施例について、データの流れの一例を示したブロック図である。図4と図6(a)に示すように、通常、フラッシュメモリ402の領域Aに書込まれたアプリケーションプログラム601をブートする場合には、ブート選択スイッチ405を“0”に設定し、領域選択スイッチ406を“0”に選択してDSP401をリセットする。リセット時にレジスタ410には領域選択スイッチ406の値“0”がサンプリングされるので、アドレスバス404の最上位ビットが“0”となり、DSP401からはフラッシュメモリ402の領域A501だけが見え、またブート選択スイッチ405が“0”に設定されているため、フラッシュメモリ402の領域Aのアプリケーションプログラム601がブートされる。

【0020】図4と図6(b)に示すように、フラッシュメモリ402の領域Aのアプリケーションプログラム601を書換える場合は、ブート選択スイッチ405を“0”、領域選択スイッチを“1”に設定してDSP401をリセットする。リセット時にレジスタ410には領域選択スイッチ406

の値“1”がサンプリングされるので、アドレスバス404の最上位ビットが“1”となり、DSP401からはフラッシュメモリ402の領域B502だけが見え、またブート選択スイッチ405が“0”に設定されているため、領域Bのメモリ書換えプログラムがブートされる。このメモリ書換えプログラム602は、ブート後にレジスタ410を“0”に書き換えてフラッシュメモリ402の領域A 501が見えるようにし、パーソナルコンピュータ408からシリアルポート409、407を介してDSP401に入力したプログラムをフラッシュメモリ402に書込む。DSP401に入力したプログラムは、レジスタ410が“0”のため、領域A501に書込まれる。

【0021】図4と図6(c)に示すように、フラッシュメモリ402の領域Bのメモリ書換えプログラム602を書換える場合は、ブート選択スイッチ405を“1”に設定してDSP401をリセットし、フラッシュメモリ402の領域Bにメモリ書換えプログラム602の書き込みを行うプログラムをパーソナルコンピュータ408からシリアルポートブートする。このプログラムは、ブート後にレジスタ410を“1”に書き換え、フラッシュメモリ402の領域B502が見えるようにし、新たなメモリ書換えプログラムをフラッシュメモリ402の領域Bに書込む。領域Bに書込むメモリ書換えプログラム602は、シリアルポート407からブートしたプログラムの中にデータテーブルとして組込まれている。

【0022】次に、メモリ書換えプログラムを内蔵しておらず外部インターフェースからブートする場合の実施例を図7によって説明する。図7は、メモリ書換えプログラムを内蔵していない場合の一実施例を説明する図である。図7において、DSP401はフラッシュメモリ402に、データバス701およびアドレスバス702で接続され、パーソナルコンピュータ408にシリアルポートインターフェース409、407で接続されている。

【0023】図8は、図7の実施例について、データの流れの一例を示したブロック図である。図7と図8(a)に示すように、通常、フラッシュメモリ402に書込まれたアプリケーションプログラム601をブートする場合には、ブート選択スイッチ703を“0”に設定してDSP401をリセットする。この場合には、フラッシュメモリ402のアプリケーションプログラム601がブートされる。図7と図8(b)に示すように、フラッシュメモリ402に書込まれているアプリケーションプログラム601を書換える場合には、ブート選択スイッチ703を“1”に設定してDSP401をリセットし、メモリ書換えプログラムをパーソナルコンピュータ408からシリアルポートブートする。このプログラムは、アプリケーションプログラム601をフラッシュメモリ402に書込む。このフラッシュメモリ402に書込むアプリケーションプログラム601は、シリアルポート407からブートするプログラムの中にデータテーブルとして組込まれている。

【0024】図4、図5、図6および図9の実施例では、アプリケーションプログラム601を書換える場合、一度フラッシュメモリ402の領域B502にメモリ書換えプログラム602を書込んでしまえば、パーソナルコンピュータ408からはメモリ書換えプログラム602を転送(図6(c)の手順)する必要はなく、図6(b)の手順で更新したいアプリケーションプログラム601だけを転送すれば良く、デバッグ作業などで頻繁にアプリケーションプログラム601を更新する場合に適している。

【0025】一方、図7、図8および図10の実施例では、アプリケーションプログラム601を書換える度に、パーソナルコンピュータ408からメモリ転送プログラムをDSP401へ転送する必要があるが、フラッシュメモリ402にメモリ書換えプログラムを書込まず、全ての領域をアプリケーションプログラム601に割当てることができるので、アプリケーションプログラム601の更新頻度が低く、プログラムサイズが大きい場合に適している。

【0026】

【発明の効果】以上のように本発明によれば、メモリを実装したまで、メモリ書換えプログラムの変更が可能である。このため、ICソケットを使用する必要がないので、装置の小型化が実現できる。

【0027】また、本発明の別の効果として、メモリ書換えプログラムのためのメモリを必要としないため、実装する素子数が低減でき、装置の小型化とともに、製造コストが低減できる。

【図面の簡単な説明】

【図1】 本発明の一実施例におけるデータの流れを説明する図。

【図2】 本発明の一実施例におけるデータの流れを説明する図。

【図3】 従来技術におけるデータの流れを説明する図。

【図4】 本発明の一実施例を説明する図。

【図5】 図4の実施例でのフラッシュメモリのデータ領域および、DSPから見たアドレスを説明する図。

【図6】 図4の実施例におけるデータの流れを説明する図。

【図7】 本発明の他の実施例を説明する図。

【図8】 図7の実施例でのデータの流れを示す図。

【図9】 ブート選択スイッチと領域選択スイッチの組合わせによるDSPのブート先を説明する図。

【図10】 ブート選択スイッチによるDSPのブート先を説明する図。

【符号の説明】

101: マイクロプロセッサ、 102: 不揮発性メモリ、
103: アプリケーションプログラム、 104: メモリの領域Aを書換えるプログラム、 105: 外部インターフェース、 106: 更新したアプリケーションプログラム、 107: 領域Bにメモリ書換えプログラムを書込むプログラ

ム、201:メモリ書換えプログラムの実行プログラムにアプリケーションプログラムをデータテーブルとして結合したもの、202:実行形式のメモリ書換えプログラム、203:アプリケーションプログラムのプログラムコード、301:ROM、302:不揮発性メモリ102を書換えるプログラム、401:DSP、402:フラッシュメモリ、403:データバス、404:アドレスバス、405:ブート選択スイッチ、406:フラッシュメモリ領*

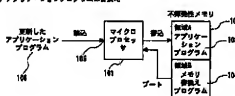
* 域選択スイッチ、407:シリアルポートインターフェース、408:パーソナルコンピュータ、409:シリアルポートインターフェース、410:レジスタ、501:領域A、502:領域B、601:アプリケーションプログラム、602:メモリ書換えプログラム、701:データバス、702:アドレスバス、703:ブート選択スイッチ、

【図1】

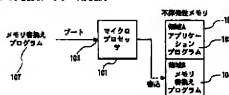
(a) 通常のブート



(b) アプリケーションプログラムの書換え

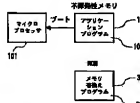


(c) メモリ書換えプログラムの書き換え

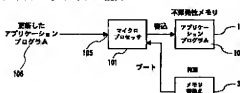


【図3】

(a) 通常のブート

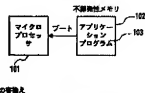


(b) アプリケーションプログラムの書換え

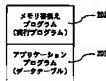
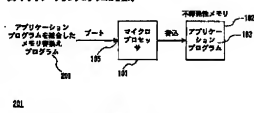


【図2】

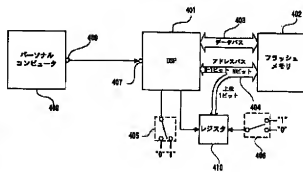
(a) 通常のブート



(b) アプリケーションプログラムの書換え



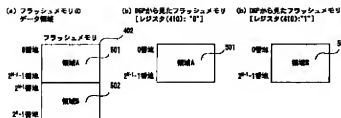
【図4】



【図9】

ブート選択 スイッチ(405)	領域選択 スイッチ(406)	DSP(401)のブート先
=0	=0	フラッシュメモリ(402) 領域A
=0	=1	フラッシュメモリ(402) 領域B
=1	=0	シリアルポートブート
=1	=1	シリアルポートブート

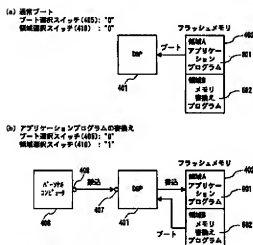
【図5】



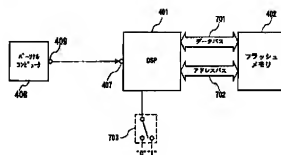
【図10】

ブート選択スイッチ(703)	DSP(401)のブート先
"0"	フラッシュメモリ(402)
"1"	シリアルポートブート

【図6】



【図7】



【図8】

